

Remarks

Applicants respectfully request reconsideration of the present application in view of the foregoing amendments and the following remarks. Claims 1, 3-5 and 7-33 are pending in the application. Claims 1, 3-5 and 7-33 are rejected. No claims have been allowed. Claims 1, 19, and 33 are independent. Claims 2 and 6 have been cancelled without prejudice. Claims 1, 19, and 33 have been amended.

Cited Art

The Action cites Davidson et al., U.S. Patent No. 6,083,276 (hereinafter “the Davidson patent”).

Claim Rejections under 35 U.S.C. § 112

The Action rejects claims 1-33 under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. The Action alleges that claims 1, 19, and 31 are “incomplete for omitting essential structural/functional cooperative relationships of elements, such omission amounting to a gap,” and more specifically for omitting “the time-related relationship gap linking” executions. [Action, at § 4, page 2.]

While Applicants do not necessarily agree that such a gap exists, for the sake of expediting prosecution, independent claims 1, 19, and 31 have been amended. The claims now recite a “first execution” and a “second execution.” Claim 1, for example, now recites:

receiving a reflection of the executable computer program during a first execution of the program;

*producing the data domain based on the domain configuration information and the reflection of the executable computer program, *the data domain representing a limited set of data values to be used as input during a subsequent second execution* of the computer program for testing the executable computer program when executed;*

targeting testing during the second execution of the executable computer program to use only values for the data structure element that fall within the data domain; and

determining whether the executable computer program behaves correctly during the second execution when executing using targeted values falling within the data domain as input.

[Emphasis added.] Applicants note that this amended claim language is consistent with the Action's observation that the Specification describes "disjoint instances" of "code execution . . . for testing and verification based on value ranges" and "the runtime in which to obtain reflection." [Action, at § 4, page 3.]

Applicants believe that, with the instant amendments, two distinct executions are clearly recited. Applicants respectfully argue that the rejection of the claims over 35 U.S.C. § 112 is now moot and request that the rejection be withdrawn.

Patentability of Claims 1-33 under 35 U.S.C. § 103(a)

The Action rejects claims 1-33 under 35 U.S.C. § 103(a) as being unpatentable over the Davidson patent. Applicants respectfully submit the claims in their present form are allowable over the cited art. The cited art does not teach or suggest at least one limitation of each of these claims. Accordingly, Applicants request that all rejections be withdrawn. Claims 1, 19, and 33 are independent.

Claim 1

Claim 1, as amended, recites, in part:

receiving a reflection of the executable computer program during a first execution of the program;

producing the data domain based on the domain configuration information and the reflection of the executable computer program, the data domain representing a limited set of data values to be used as input during a subsequent second execution of the computer program for testing the executable computer program when executed;

targeting testing during the second execution of the executable computer program to use only values for the data structure element that fall within the data domain; and

determining whether the executable computer program behaves correctly during the second execution when executing using targeted values falling within the data domain as input.

[Emphasis added.] Support for the above-emphasized language can be found, for example, as the Action notes (at §4, page 3 of the Action) at pages 10 and 18 of the Application.

The Davidson patent does not teach or suggest "receiving a reflection of [an] executable computer program during a first execution," "targeting testing during the second execution of the executable computer program" or "determining whether the executable computer program

behaves correctly during the second execution when executing using targeted values” because the Davidson patent’s actions are performed in order to generate a program, and therefore are done on an incomplete program. In its rejection of claim 1, the Action acknowledges that these elements of the claim are not described in the Davidson patent:

[]Davidson does not explicitly teach obtaining reflection of an executable program during its execution, the program to be used to target efficient testing of behavior of the executable program during its execution, the executable program having been compiled into executable form.

[Action, at § 6, page 6.] However, the Action makes two arguments:

1) The Action argues, in portions (B) and (C) of the Response to Arguments section, that, because the Action had determined the claims to be indefinite under 35 U.S.C. § 112, that the various “execution” language did not hold patentable weight. As to this argument, Applicants have amended the claims to address the § 112 rejection, as discussed above. Applicants believe that the language has patentable weight and should be afforded such weight during subsequent examination. Applicants respectfully request that the claim language be reconsidered relative to the Davidson patent in light of this amendment. In particular, Applicants note that, for reasons discussed at great length in the previous Amendment of February 28, 2008, the Davidson patent does not teach or suggest “targeting testing during the second execution of the executable computer program” after “receiving a reflection of [an] executable computer program during a first execution” as recited in claim 1.

2) The Action also argues that it would have been obvious to modify the Davidson patent to “implement Sun Microsystems API in testing a target bean code as endeavored by Davidson, so that reflection data is obtained using such runtime API . . . and to test such target compiled bean.” [Action, at §6, page 7.] In doing so, the Action argues that the Davidson patent “discloses an execution environment . . . to dynamically determine . . . some expected value or range of Java elements . . . , such runtime determination being based on . . . domain data,” citing to a “Generate Error” step 458 from Figure 4D of the Davidson patent. [See, Action, at § 6, page 6.] The Action also relies on this argument as well in its Response to Arguments sections (A) and (B).

Applicants respectfully traverse this argument. Step 458, which states “Generate Error” is part of Figure 4D of the Davidson patent. The Davidson patent states very clearly, however,

that the process of Figure 4D is still performed to generate a program, and is not performed while the target program is executing:

Referring now to FIG. 4D, there is shown a method of processing 406 the components 212 to launch the component-based application 214 in accordance with a preferred embodiment of the present invention.

[Davidson patent, at column 27, lines 1-4; emphasis added.] The Davidson patent clearly spells out that this process is performed *prior to execution* further down in column 27:

In one embodiment of the invention, after all of the components 212 have been processed, the init() and start() methods of the root component 212 are invoked in order to begin actual execution of the application 214.

[Davidson patent, at column 27, lines 43-46; emphasis added.] As such, even if, for the sake of Argument, the process of Figure 4D in the Davidson patent were considered to read on some form of “testing,” the Davidson patent’s processing of components does not teach or suggest, and in fact teaches away from “receiving a reflection of [an] executable computer program during a first execution” and “determining whether the executable computer program behaves correctly during the second execution” because the Davidson patent’s processing of components is performed *before beginning execution*. In fact, Applicants note that the process of Figure 4D of the Davidson patent is simply an extension of the process of figure 4C, which Applicants previously argued was performed in order to generate a program, and is therefore done on an incomplete program, which is not executable.

For at least these reasons, the Davidson patent does not teach or suggest at least the above-quoted language of claim 1. Claim 1, as well as claims 3-5 and 7-18, which depend from claim 1, are thus allowable, and Applicants request their allowance. Applicants will not belabor the merits of the separate patentability of dependent claims 3-5 and 7-18.

Claim 19

Claim 19 recites:

using a reflection of the executable computer program produced during a first execution of the program to produce the data domain for the data structure element according to the domain configuration information, the data domain representing a limited set of data values to be used as input for testing the executable computer program when executed; and

controlling testing during a second execution of the executable computer program to use only values for the data structure element that fall within the data domain.

[Emphasis added.] For at least the reasons discussed above with regard to claim 1, the Davidson patent does not teach or suggest the above-quoted language of claim 19. Claim 19, as well as claims 20-32, which depend from claim 19, are thus allowable, and Applicants request their allowance. Applicants will not belabor the merits of the separate patentability of dependent claims 20-32.

Claim 33

Claim 33 recites:

means for reading, on the computer apparatus, a reflection of the executable computer program during a first execution of the program;

means for processing, on the computer apparatus, the domain configuration information and the reflection to produce and output the data domains corresponding to the data structure elements, the data domains representing a limited set of data values to be used as input during a second execution of the computer program for testing the computer program when executed; and

means for limiting testing during the second execution of the executable computer program to use only values for the data structure element that fall within the data domain.

[Emphasis added.] For at least the reasons discussed above with regard to claim 1, the Davidson patent does not teach or suggest the above-quoted language of claim 33. Claim 33 is thus allowable, and Applicants request its allowance.

Interview Request

If the claims are not found by the Examiner to be allowable, the Examiner is requested to call the undersigned attorney to set up an interview to discuss this application.

Conclusion

The claims in their present form should be allowable. Such action is respectfully requested.

Respectfully submitted,

KLARQUIST SPARKMAN, LLP

One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Telephone: (503) 595-5300
Facsimile: (503) 595-5301

By /Kyle B. Rinehart/
Kyle B. Rinehart
Registration No. 47,027